数学启发算法求解单源设施区位及其变种问题

孔云峰[1,2]

1 河南大学黄河中下游数字地理技术教育部重点实验室，河南 开封，475000；2 河南大学地理与环境学院，河南 开封，475000

摘要：考虑设施服务区空间连续和设施数量限制，对经典的单源设施区位问题(SSCFLP)进行扩展，形成变种问题：设施服务区空间连续 SSCFLP(CFLSAP)、设施数量限制的 SSCFLP (SSCKFLP)和设施服务区空间连续及数量限制的 SSCFLP(CKFLSAP)。针对 SSCFLP 及其变种问题，提出了一个数学启发算法。该算法从一个初始解开始，迭代地对当前解进行超大邻域搜索改进，直到若干次尝试不能该改进当前解为止。超大邻域定义步骤如下：随机选择一个客户，从当前解中选择客户附近 Q 个最邻近设施及其客户，再挑选这些客户的最邻近候选设施。使用所选择候选设施子集和客户子集，构建子问题数学模型，求解模型，并使用模型解更新当前解。构造 2 组案例数据测试本文算法，结果表明：数学启发算法能够有效地求解 SSCFLP 及其变种问题，求解结果极其接近问题最优解或目标值下界，相对差异为0.01%(SSCFLP)、0.22%(CFLSAP)、0.00%(SSCKFLP)和 0.08%(CKFLSAP)。另外，SSCFLP 或 SSCKFLP 增加设施服务区空间连续约束后，目标值增加幅度不大，但最优设施区位可能发生变化。

关键词：单源设施区位问题；空间连续；数学启发算法；超大规模邻域搜索。

1 引言

设施区位问题(FLP)是一大类寻找最优设施位置的问题。该问题广泛应用于学校、医疗、公安、养老、应急、物流等公共或商业设施选址决策。根据应用场景的不同，FLP 类型众多：连续空间区位或离散区位、有无设施容量限制、需求是否允许拆分指派、是否考虑设施成本、是否指定设施数量等，规划目标划分为效率目标、公平目标等。根据设施服务与需求是否存在不确定性，又可将 FLP 划分为确定性 FLP 和随机性 FLP。其中，单源 CFLP 问题(SSCFLP)是需求不允许拆分指派的、有容量约束的、顾及设施成本的离散区位问题，求解难度较高，也是近年关注度最高的区位问题之一。

令集合I为潜在设施区位，集合J为客户区位，设施i ($i\in I$)有最大服务容量s_i和固定建设成本为f_i，客户j ($j\in J$)有服务需求d_i，被设施i服务的费用为c_{ij}。定义布尔型决策变量y_i为是否在区位i建设施，布尔型决策变量x_{ij}为是否指派设施i服务客户j，SSCFLP数学模型如下：

$$\text{Minimize} \sum_{i\in I} f_i + \sum_{i\in I}\sum_{j\in J} c_{ij}x_{ij} \quad (1)$$

$$\text{Subject to: } \sum_{i\in I} x_{ij} = 1, \forall j\in J \quad (2)$$

项目资助：国家自然科学基金项目(41871307) ；

通讯作者：孔云峰，博士，教授，主要从事空间分析、空间优化等研究。yfkong@henu.edu.cn

$$\sum_{j\in J} d_j x_{ij} \le s_i y_i, \forall i \in I \quad (3)$$

$$y_i = \{0,1\}, \forall i \in I \quad (4)$$

$$x_{ij} = \{0,1\}, \forall i \in I, j \in J \quad (5)$$

模型中，目标函数(1)最小化设施固定成本和获取服务成本。约束(2)确保每个客户由唯一的设施提供服务；约束(3)要求指派给每个设施的客户需求量不超过其最大服务容量。约束(4)和(5)定义两类布尔型决策变量。

求解 SSCFLP 问题的方法划分为两大类：精确算法和启发式算法(Basu et al. 2015; Ulukan & Demircioğlu 2015)。精确算法包括：分枝定界(Neebe & Rao 1983; Holmberg et al. 1999)、列生成(Díaz & Fernández 2002)、Cut-and-Solve (Yang et al. 2012; Gadegaard et al. 2018)等。启发式算法包括：拉格朗日松弛启发(LH)(Barcelo & Casanova 1984; Klincewicz & Luss 1986; Beasley 1993; Sridharan 1993; Agar & Salhi 1998; Hindi & Pienkosz 1999; Cortinhal & Captivo 2003)、禁忌搜索(Filho & Galvao1998; Delmaire et al. 1999; Cortinhal & Captivo 2003)、超大规模邻域搜索(VLNS) (Ahuja et al. 2004; Tran et al. 2017)、分散搜索(Contreras & Diaz 2008)、核搜索(Kernel Search)(Guastaroba & Speranza 2014)、"廊道"方法(Corridor Method)(Caserta & Voß 2020)、松弛自适应记忆规划(RAMP)(Oliveira et al. 2020)等。因 SSCFLP 问题具有极高的计算复杂度，精确算法难以高效求解规模较大的案例；LH 算法速度快，但求解质量不能保证；超大规模邻域搜索、核搜索和"廊道"方法等元启发式算法在求解质量方面具有竞争力，但计算时间仍然偏长。

在 SSCFLP 应用实践中，也存在需求特定需求，如拟建设施数量约束(Aardal et al. 2015; Wang 2017)、要求每个设施服务区空间连续等。例如，某一区域进行义务教育学校布局规划，需要考虑学校建设成本、学校数量要求，在确定学校区位的同时划定其学区。某一城市进行基层卫生服务中心布局规划，遵照国家行业规范，每个中心覆盖一定的空间范围，满足 15 分钟出行要求，且考虑中心的服务能力。为求解这些问题，有必要扩展 SSCFLP 模型，增加这两个约束条件。在分区问题研究中，空间连续约束有三个建模方法：生成树模型、次序模型和流模型(Shirabe 2009; Duque et al. 2011)。树模型中约束条件数量庞大，仅适用于小规模案例(结点数量小于 50)；次序模型和流模型的约束条件相对较少。针对 p-regions 问题，三个模型在 3 小时内最多能够求解 49 个空间单元的案例。为 SSCFLP 扩展空间连续约束，精确方法的求解能力仍是未知数。

基于以上讨论，本文扩展 SSCFLP 模型，考虑设施服务区的空间连续性以及设施数量要求，设计一个数学启发算法求解 SSCFLP 及其扩展问题，并测试算法的求解性能。

2 问题定义

一个特定的地理区域，包含 n 个空间单元。每一个空间单元 j 具有属性 d_j，表示空间单元 j 的服务需求数量；n 个单元中有 m (m≤**n**)个单元适合建设设施，作为候选设施区位，设施 i 服务容量为 s_i，固定建设成本为 f_i。单元 i 被设施 j 服务的费用为变量 c_{ij}(空间单元 i 与 j 之间的单位交通成本与需求量 d_j 的乘积)。令集合 I 表示 m 个候选设施单元，集合 J 为 n 个需求单元($I \subseteq J$)，则 SSCFLP 模型(1)-(5)也适用于该区域内设施选址。

要求设施服务区空间连续，需要在 SSCFLP 模型中增加空间连续约束。将地理空间表达为一个网络图，即将空间单元作为结点，相邻空间单元相连接作为网络边，形成一个无向网络图。基于网络图，可采用“网络流”概念表达分区空间连续性(Duque et al. 2011)。在一个设施服务区内，每个单元产生一个单位的流量，通过网络边流动，最终汇入到区内设施单元。设施单元不产生流量，但汇入最高达 n-K 单位的流量(K 为设施数量)。产流和汇流单元所构成的区域是一个空间连续的区域。令集合 N_j 表示单元 j 的相邻单元，整型变量 f_{ijk} 表示设施 i 服务区内单元 j 到单元 k 的流量，则可构建空间连续约束条件：

$$f_{ijk} \leq n * x_{ij}, \forall i \in I, j \in J, k \in N_j \quad (6)$$

$$f_{ijk} \leq n * x_{ik}, \forall i \in I, j \in J, k \in N_j \quad (7)$$

$$\sum_{k \in N_j} f_{ijk} - \sum_{k \in N_j} f_{ikj} \geq x_{ij}, \forall i \in I, j \in J \backslash i \quad (8)$$

$$f_{ijk} \geq 0, \forall i \in I, j \in J, k \in N_j \quad (9)$$

条件(6)和(7)确保属于同一个服务区的两个相邻单元间才可能产生流。对于设施 i 不服务单元 j($x_{ij} = 0$)，则单元 j 在 i 服务区中不会产生流f_{ijk}=0，也不会有汇流f_{ikj}=0；当设施 i 服务单元 j 在($x_{ij} = 1$)时，可能会有进出单元 j 的流。若单元 j 不是汇集单元(设施单元)，约束(8)保证该单元至少产生一个单位的流。约束(6)-(8)将使每一个设施服务区中的产流汇集到设施单元。一个设施服务区内，每个空间单元必需产生流且流向邻域单元，最终汇集到设施单元，这使得一个设施服务区内任意两个单元都是连通的，从而保证了服务区空间连续。

需要指定设施数量时，增加条件(10)，指定设施数量为 K。

$$\sum_{i \in I} y_i = K \quad (10)$$

基于以上约束，有三个 SSCFLP 扩展模型：①设施服务区空间连续约束的 SSCFLP，记为 CFLSAP，模型包括式(1)-(9)；②设施数量约束的 SSCFLP，记为 SSCKFLP，模型包括式(1)-(5)和(10)；③设施服务区空间连续约束和设施数量约束的 SSCFLP，记为 CKFLSAP，模型包括式(1)-(10)。

2 数学启发算法

局部搜索算法使用邻域搜索用于渐进改进当前解，广泛应用于离散优化问题求解。区位问题中，设施的变动必然会引起需求指派变化，使得邻域搜索较为复杂。因此，区位问题算法设计的一个核心问题是：如何有效地移动设施并更新指派，寻找到可行的、更节约的邻域解。因设施容量约束，涉及较多设施的超大规模邻域搜索才有较高可能性改进当前解。然而，SSCFLP 的超大规模邻域结构定义复杂，邻域搜索的计算复杂度也很高，存在指数级数量的客户移动移动和设施移动(Ahuja et al. 2004; Tran et al. 2017)。

本文拟采用数学模型求解超大邻域。算法原理如下：①使用 LH 方法(Holmberg et al. 1999)、线性松弛或启发式算法获得问题初始解；②从当前解中随机选择一个超大邻域，获得邻域内需求点、当前设施和候选设施集合；③构造邻域内子问题模型并求解，更新

当前解；④重复执行步骤②和③，直到若干次尝试不能更新当前解。步骤②构造一个大邻域，步骤③用于发现邻域内的最优解。基于这一思路，数学启发算法示意如下：

参数:连续未更新最好解循环数(*mloops*)

```
1. s= GenerateInitialSolution();
2. notImpr=0;
3. While notImpr < mloops:
4.      Select a large neighborhood I* and J* from solution s;
5.      s*= SolveSubProblem(I*, J*);
6.      s'=UpdateSolution(s,s*);
7.      s''=VNDSearch(s')
8.      If f(s'')<f(s): s=s'', notImpr=0;
9.          else: notImpr+=1;
10. Output(s).
```

以上算法中，步骤(1)获得初始解。因很多案例难以获得可行的初始解，本文按以下顺序考虑解的可行性：指派约束、设施数量约束、空间连续约束和设施容量约束。考虑到设施容量约束难以满足，本文算法允许设施容量超限，但在目标函数中对超限量进行惩罚。增加整型决策变量$H_i = \{0,1,2 \ldots\}, \forall i \in I$，表示超出设施 *i* 服务能力的量，容量约束和目标函数可改为：

$$\sum_{j\in J} d_i x_{ij} \leq s_i y_i + H_i, \forall i \in I \tag{11}$$

$$Minimize\colon\ \sum_{i\in I} f_i + \sum_{i\in I}\sum_{j\in J} c_{ij} x_{ij} + \alpha \sum_{i\in I} H_i \tag{12}$$

通过以上模型修改，允许设施容量超限，容易产生初始解。设置设施容量超限惩罚系数α足够大，后续算法逐步减小H_i值，直到变量值为零，从而使容量约束条件(3)得到满足。这一设施容量软约束也使 MIP 优化器快速获得可行解，有利于模型精确求解。

初始解生成方法可选择：拉格朗日松弛启发、线性松弛启发、ADD 或 DROP 启发方法。拉格朗日松弛启发算法(Holmberg et al. 1999)可直接求解 SSCFLP 案例，通过调整设施数量可用于求解 SSCKFLP。线性松弛启发是将修改布尔变量x_{ij}为[0,1]连续变量，使用优化器对模型求解，对求解结果进行修复。ADD 启发或 DROP 启发是经典的构造算法，以贪心方式获得可行解。考虑服务区连续时，上述算法结果需要进行连续性修复。

算法步骤(4)从当前解中随机选择一个超大邻域，包括设施子集 *I**和需求子集 *J**。选择邻域是算法设计的一个关键点，步骤如下：①随机选择一个需求点，以该点为中心，从当前解设施集合中自近而远地选择 *Q* 个设施；②指派给 *Q* 个设施的需求点构成需求点子集 *J**；③子集 *J**中每个需求点最近的设施以及已选择的 *Q* 个设施构成子集 *I**。以上步骤所选择邻域的大小受 *Q* 值影响，*Q* 值取值范围为[Q_{min}, Q_{max}]，其中，Q_{min} = min(*L*/2,7)，Q_{max} =min(*L*,10)，*L* 为当前解中的设施数量。当 *L*≤10 时，*Q* 处于区间[*L*/2, *L*]；当 11≤*L*≤13 时，*Q* 处于区间[*L*/2,10]；当 *L*≥14 时，*Q* 处于区间[7,10]。

邻域规模影响算法性能：若邻域太小，当前设施区位往往难以更新；若邻域过大，模型计算效率低。当集合 *I**的规模偏大时，可能会使模型计算效率低下。为此，需要精

简一部分候选设施，再进行子问题建模和求解。精简方法：从 *I**选择 *Q* 个当前解中已使用设施，再从其余设施中随机选择 *Q* 个设施，更新集合 *I**，使其包括 2*Q* 个设施。

步骤(5)使用设施子集 *I**和需求子集 *J**构造子问题模型，并使用 MIP 优化器求解。针对 SSCFLP、CFLSAP、SSCKFLP 和 CKFLSAP 四个问题，前两者构造 SSCFLP 模型，后两者构造 SSCKFLP 模型，保证条件(10)得到满足。

步骤(6)使用局部解 *s**更新当前解，获得新解 *s'*。对于 CFLSAP 和 CKFLSAP 问题，新解 *s'*可能不满足空间连续要求，此时需要进行对新解进行修复。

步骤(6)修复过程通常会使目标值较差，步骤(7)使用可变邻域下降(VND)算法改进当前解。VND 算法中采用 1 单元移动和 2 单元移动算子(Kong et al. 2017)，获得局部最优解。

步骤(8)尝试更新当前解。若新解 *s''*优于当前解 *s*，更新当前解；否则，维持当前解不变。

迭代执行步骤(4)-(9)，当连续若干次循环未能更新当前解，算法终止。

3 算法测试

3.1 案例设计

本文基于两个典型区域构造测试案例。区域 ZY 为城区，约 13.4 平方千米，包含 326 个空间单元，有 15 所小学、3783 名学生；区域 ZY 为县域，约 1000 平方千米，包含 1276 个空间单元，有 20 个乡镇街道办，人口 819812 人。区域 ZY 将学生人数作为需求量，在现有学校基础上手工增加部分点位作为潜在设施区位，设施容量参照现有学校均值。区域 GY 将人口数量作为需求量，在现有乡镇街道办所在地基础上手工增加部分区位作为潜在设施区位，设施容量参考现有乡镇街道办人口数量均值。图 1 为案例区域服务需求和候选服务设施的空间分布示意图，浅灰色圆圈表示需求量，星形符号表示潜在设施。

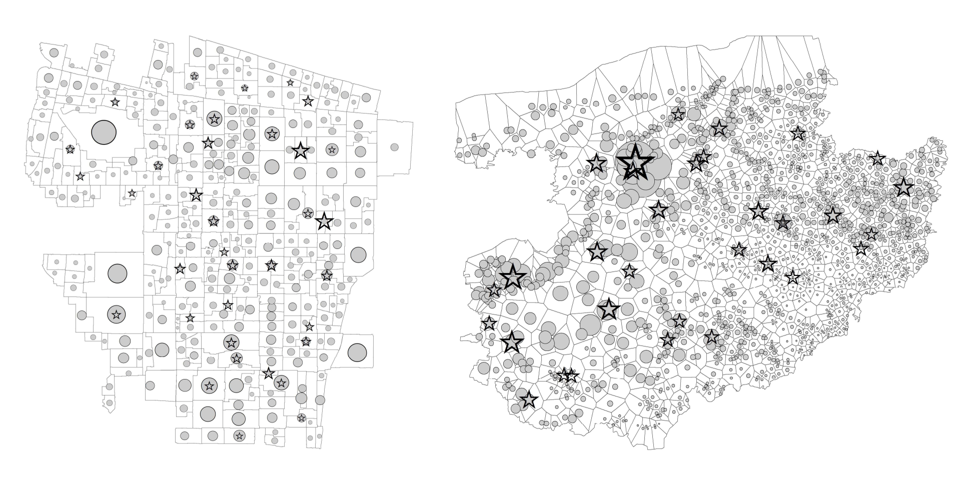


图 1 案例区域 ZY(左)和 GY(右)

Figure 1 The study areas ZY (lest) and GY (right)

首先，基于两个地理区域数据构造两个测试案例。每个区域中，将空间单元中心点位置作为需求点位置和候选设施位置；假设设施 i 服务客户 j 的费用$c_{ij} = 1.0 * d_{ij} * d_j$，其中$d_{ij}$为设施 i 和客户 j 间的欧氏距离，单位为公里，d_j为客户 j 的需求量。设施建设成本$f_i = (\mu + \varepsilon_i)s_i$，$s_i$为设施服务容量，$\varepsilon_i$为一定范围内的随机数。对于区域 ZY，取$\mu$=0.8、$\varepsilon_i$为区间[-0.1, 0.1]内的一个随机数；对于区域 ZY，取μ=1.8、ε_i为区间[-0.2, 0.2]内的一个随机数。参数μ的取值经过多次尝试，使 SSCFLP 总成本中设施成本比例不能太高，也不能太低，处于 30%~70%之间。

其次，基于 ZY 和 GY 案例数据构造测试案例集。针对每个案例，扩大设施服务能力为 1.2 倍和 1.4 倍，增加设施成本为 1.1、1.2、1.3 和 1.4 倍。设施容量与设施成本组合，每个区域上共有 15 个案例。案例基本信息见表 1。表中，供需比等于所有候选设施的总容量与总需求量之比。已有研究表明：供需比是案例求解难度的影响因素之一。表中，设施成本容量比等于候选设施的总成本与总容量之比。案例数据可从 https://github.com/yfkong/unified 下载。

表 1 测试案例集

Table 1 The benchmark instances

案例	$\|I\|$	$\|J\|$	供需比	设施成本容量比	案例	$\|I\|$	$\|J\|$	供需比	设施成本容量比
zya1	36	326	2.37	0.80	gya1	33	1276	1.61	1.83
zya2	36	326	2.37	0.88	gya2	33	1276	1.61	2.02
zya3	36	326	2.37	0.96	gya3	33	1276	1.61	2.20
zya4	36	326	2.37	1.04	gya4	33	1276	1.61	2.38
zya5	36	326	2.37	1.12	gya5	33	1276	1.61	2.56
zyb1	36	326	2.84	0.67	gyb1	33	1276	1.93	1.53
zyb2	36	326	2.84	0.74	gyb2	33	1276	1.93	1.68
zyb3	36	326	2.84	0.80	gyb3	33	1276	1.93	1.82
zyb4	36	326	2.84	0.87	gyb4	33	1276	1.93	1.98
zyb5	36	326	2.84	0.94	gyb5	33	1276	1.93	2.14
zyc1	36	326	3.32	0.57	gyc1	33	1276	2.26	1.31
zyc2	36	326	3.32	0.63	gyc2	33	1276	2.26	1.44
zyc3	36	326	3.32	0.69	gyc3	33	1276	2.26	1.57
zyc4	36	326	3.32	0.75	gyc4	33	1276	2.26	1.70
zyc5	36	326	3.32	0.80	gyc5	33	1276	2.26	1.83

3.2 案例测试

本文数学启发算法使用 Python 程序设计语言编程实现。算法步骤如下：读数据，使用 LH 方法获得初始解，迭代执行邻域选择、子问题求解和更新当前解，直到满足终止条件。子问题求解环节，使用 PuLP(https://github.com/coin-or/pulp)线性规划建模工具完成 SSCFLP 或 SSCKFLP 子问题模型建构，再调用 IBM CPLEX 12.6 优化器完成模型求解。为提升计算速度，算法在 PyPy7 (https://www.pypy.org)环境中运行。实验计算环境为：HP 桌面计算机，配置 Intel Core I7-6700 CPU 3.40-GHz 和 8GB 内存，Windows 10 操作系统。本文算法可从网页 https://github.com/yfkong/unified 下载，按要求安装软件即可运行。

使用案例数据完成 4 个问题的测试。所有案例进行测试 SSCFLP、CFLSAP 问题求解；针对 zya1 和 gya1 案例，测试 SSCKFLP、CKFLSAP 问题求解，案例 zya1 设置设施数量 K=13-22，案例 gya1 设置设施数量 K=18-30。

使用两种方法进行问题求解：精确方法和本文数学启发方法。前者根据本文模型构造案例模型，使用 CPLEX 12.6 优化器求解。记录 CPLEX 计算结果中的案例目标值下界、目标值上界及计算时间。数学启发算法设置参数 *mloops* 为 100。一般来说，该参数数值越大，所需计算时间也越长。记录计算结果的目标值、目标值与下界差距和计算时间。

3.3 计算结果

计算结果分别见表 2、表 3、表 4 和表 5。表中，LB 为案例目标值下界，UB 为案例目标值上界，Gap 为 LB 和 UB 的相对差异。若两者相等表明(Gap=0)，表明 CPLEX 找到了案例最优解。表中最右三栏分别为数学启发算法获得的案例目标值、目标值与下界相对差异 Gap，以及计算时间。

表 2 SSCFLP 计算结果

Table 2 Detailed SSCFLP solutions

案例	CPLEX				数学启发算法		
	LB	UB	Gap/%	时间/s	目标值	Gap/%	时间/s
zya1	4428.23	4428.23	0.00	877.53	4428.23	0.00	62.23
zya2	4722.23	4722.23	0.00	780.86	4722.23	0.00	72.15
zya3	5025.23	5025.23	0.00	988.81	5025.57	0.01	58.55
zya4	5327.60	5327.60	0.00	2042.05	5329.28	0.03	54.16
zya5	5626.94	5626.94	0.00	1458.30	5628.48	0.03	88.73
zyb1	3999.99	3999.99	0.00	6024.45	4006.75	0.17	130.50
zyb2	4239.99	4239.99	0.00	3368.16	4239.99	0.00	162.80
zyb3	4488.99	4488.99	0.00	7106.34	4488.99	0.00	136.51
zyb4	4736.99	4736.99	0.00	2676.81	4736.99	0.00	86.87
zyb5	4982.99	4982.99	0.00	2662.05	4982.99	0.00	78.30
zyc1	3732.85	3732.85	0.00	227.04	3732.85	0.00	105.13
zyc2	3940.85	3940.85	0.00	237.91	3940.85	0.00	89.73
zyc3	4158.85	4158.85	0.00	457.83	4158.85	0.00	94.61
zyc4	4374.85	4374.85	0.00	293.48	4374.85	0.00	96.94
zyc5	4589.85	4589.85	0.00	758.59	4590.17	0.01	118.67
gya1	3498793.22	3499012.55	0.01	7200.00	3499012.55	0.01	183.54
gya2	3657078.11	3657323.63	0.01	7200.00	3657364.63	0.01	270.68
gya3	3812527.81	3812605.72	0.00	7200.00	3812613.41	0.00	207.99
gya4	3966893.30	3966921.73	0.00	7200.00	3966941.43	0.00	221.26
gya5	4119130.39	4119543.12	0.01	7200.00	4119523.42	0.01	204.48
gyb1	3433138.33	3433223.99	0.00	7200.00	3433223.99	0.00	122.99
gyb2	3573531.57	3573616.42	0.00	7200.00	3573616.42	0.00	151.29
gyb3	3712631.49	3712713.37	0.00	7200.00	3717341.49	0.13	184.42
gyb4	3847355.36	3847386.80	0.00	7200.00	3847386.80	0.00	194.18
gyb5	3977990.12	3977990.12	0.00	2681.64	3978214.70	0.01	190.17
gyc1	3351292.23	3351292.23	0.00	587.87	3351292.23	0.00	203.33
gyc2	3475209.95	3475209.95	0.00	863.11	3475209.95	0.00	144.90
gyc3	3599127.67	3599127.67	0.00	1157.06	3599127.67	0.00	157.37
gyc4	3723045.39	3723045.39	0.00	137.38	3724065.87	0.03	281.80
gyc5	3840616.19	3840616.19	0.00	5198.12	3840702.03	0.00	313.58

表 2 中 SSCFLP 计算结果表明：CPLEX 能够求解本文案例，获得最优解或高质量近似解(CPLEX MIPGap<0.01%)，但计算时间差异较大，最快 2 分钟，最慢 2 小时未发现最优解。数学启发算法计算速度很高，各案例需要 1~6 分钟，计算结果质量很高，12 个案例获得最优解，其余案例除 zyb1 和 gyb3 外，目标值 Gap 小于 0.01%。

表 3 CFLSAP 计算结果表明：CPLEX 求解本文案例，均获得满意的结果，平均 Gap 值为 0.20%，5 个案例在 2 小时时间内获得最优解；与表 2 相比，增加空间连续约束后，CPLEX 计算时间显著增加。数学启发算法需要 1-5 分钟时间求解各案例，平均 Gap 值为 0.22%，发现 2 个案例最优解，12 个案例求解质量高于 CPLEX 结果。

表 3 CFLSAP 计算结果

Table 3 Detailed CFLSAP solutions

	CPLEX				MH		
案例	LB	UB	Gap/%	时间/s	目标值	Gap/%	时间/s
zya1	4435.21	4446.32	0.25	7200.00	4443.21	0.18	102.00
zya2	4727.23	4740.69	0.28	7200.00	4740.18	0.27	211.85
zya3	5031.18	5045.69	0.29	7200.00	5052.03	0.41	149.08
zya4	5335.83	5344.66	0.17	7200.00	5365.55	0.56	62.45
zya5	5634.87	5652.69	0.32	7200.00	5646.41	0.20	79.93
zyb1	4014.58	4014.58	0.00	3034.54	4015.33	0.02	92.11
zyb2	4258.35	4263.26	0.12	7200.00	4263.08	0.11	233.67
zyb3	4502.29	4519.58	0.38	7200.00	4528.97	0.59	111.21
zyb4	4751.57	4778.24	0.56	7200.00	4779.08	0.58	139.89
zyb5	4994.83	5032.58	0.75	7200.00	5030.37	0.71	173.50
zyc1	3736.16	3736.16	0.00	5988.96	3736.44	0.01	76.10
zyc2	3941.78	3944.28	0.06	7200.00	3945.2	0.09	74.92
zyc3	4161.52	4162.16	0.02	7200.00	4162.44	0.02	63.36
zyc4	4372.17	4378.16	0.14	7200.00	4378.44	0.14	98.72
zyc5	4593.16	4593.16	0.00	6698.44	4594.31	0.02	65.02
gya1	3494350	3499875	0.16	7200.00	3499833	0.16	137.27
gya2	3656614	3658269	0.05	7200.00	3658247	0.04	286.00
gya3	3811904	3814304	0.06	7200.00	3815345	0.09	102.43
gya4	3950935	3968368	0.44	7200.00	3968574	0.45	141.11
gya5	4111963	4121919	0.24	7200.00	4120543	0.21	104.26
gyb1	3433408	3433408	0.00	5147.80	3433408	0.00	113.19
gyb2	3573800	3573800	0.00	6247.53	3573800	0.00	135.53
gyb3	3712195	3713060	0.02	7200.00	3712897	0.02	156.98
gyb4	3844879	3852609	0.21	7200.00	3865510	0.54	130.93
gyb5	3975431	3991801	0.41	7200.00	3978666	0.08	223.14
gyc1	3353103	3355378	0.07	7200.00	3357712	0.14	110.18
gyc2	3475415	3482023	0.19	7200.00	3482121	0.19	117.73
gyc3	3599353	3612109	0.35	7200.00	3605828	0.18	129.13
gyc4	3723704	3730328	0.17	7200.00	3731717	0.22	239.15
gyc5	3839448	3852543	0.34	7200.00	3850299	0.28	253.08

表 4 中 SSCKFLP 计算结果表明：增加设施数量约束后，CPLEX 计算时间大幅降低，20 个案例能够获得最优解，其余 3 个案例获得高质量解(Gap<0.02%)，18 个案例仅需要 2 分钟计算时间。数学启发算法求解质量高，发现 11 个案例的最优解，1 个案例目标值优于 CPLEX 结果，Gap 均值为 0.00%，最大 0.04%。数学启发算法计算时间处于 1~6 分钟。

表 4 SSCKFLP 计算结果

Table 4 Detailed SSCKFLP solutions

案例	K	CPLEX				数学启发算法		
		LB	UB	Gap/%	时间/s	目标值	Gap/%	时间/s
zya1	13	4846.88	4846.88	0.00	140.14	4848.83	0.04	101.15
zya1	14	4740.14	4740.14	0.00	84.52	4740.42	0.01	107.92
zya1	15	4644.42	4644.42	0.00	26.14	4644.47	0.00	88.33
zya1	16	4549.83	4549.83	0.00	613.82	4549.83	0.00	193.39
zya1	17	4479.05	4479.05	0.00	7.91	4479.05	0.00	75.13
zya1	18	4459.38	4459.38	0.00	56.88	4459.38	0.00	56.50
zya1	19	4428.23	4428.23	0.00	585.82	4428.57	0.01	62.06
zya1	20	4441.19	4441.19	0.00	19.77	4441.19	0.00	74.95
zya1	21	4490.37	4490.37	0.00	4.83	4490.37	0.00	34.00
zya1	22	4617.61	4617.61	0.00	73.34	4617.88	0.01	44.80
gya1	18	3516721.82	3517274.15	0.02	7200.00	3517195.85	0.01	329.68
gya1	19	3501647.12	3501975.32	0.01	7200.00	3501975.32	0.01	203.00
gya1	20	3498731.49	3499012.55	0.01	7200.00	3499012.55	0.01	140.92
gya1	21	3499745.27	3499745.27	0.00	122.78	3499745.27	0.00	126.22
gya1	22	3509618.91	3509618.91	0.00	63.27	3509618.91	0.00	87.52
gya1	23	3531467.24	3531467.24	0.00	50.42	3531467.24	0.00	76.24
gya1	24	3554134.39	3554134.39	0.00	40.91	3554134.39	0.00	88.87
gya1	25	3584373.45	3584373.45	0.00	15.19	3584373.45	0.00	56.50
gya1	26	3621833.78	3621833.78	0.00	14.58	3621840.41	0.00	44.42
gya1	27	3663089.92	3663089.92	0.00	22.06	3663089.92	0.00	46.83
gya1	28	3715849.01	3715849.01	0.00	18.59	3715872.84	0.00	47.81
gya1	29	3772555.65	3772555.65	0.00	19.95	3772579.48	0.00	51.00
gya1	30	3830365.47	3830365.47	0.00	14.66	3830366.59	0.00	51.07

表 5 CKFLSAP 案例计算结果表明：增加空间连续约束后，CPLEX 仍能够高质量求解本文案例，但计算时间大幅增加，16 个案例获得最优解，仅 2 个案例的 Gap 值大于 0.10%。数学启发算法能够快速获得高质量解，共发现 6 个案例的最优解，4 个案例优于 CPLEX 计算结果。

表 5 CKFLSAP 计算结果

Table 5 Detailed CKFLSAP solutions

案例	K	CPLEX				数学启发算法		
		LB	UB	Gap/%	时间/s	目标值	Gap/%	时间/s
zya1	13	4855.13	4855.13	0.00	4816.65	4870.93	0.33	67.62
zya1	14	4745.68	4745.68	0.00	3474.80	4760.25	0.31	138.40
zya1	15	4650.36	4650.36	0.00	2200.54	4652.13	0.04	84.74
zya1	16	4562.21	4562.59	0.01	7200.00	4570.24	0.18	80.41
zya1	17	4486.15	4486.15	0.00	2200.07	4486.15	0.00	65.02
zya1	18	4469.29	4469.29	0.00	5032.55	4475.10	0.13	67.40
zya1	19	4438.07	4441.13	0.07	7200.00	4449.64	0.26	51.51
zya1	20	4446.32	4446.32	0.00	2078.38	4447.13	0.02	59.24
zya1	21	4500.53	4500.53	0.00	394.76	4506.77	0.14	24.85

zya1	22	4619.83	4620.24	0.01	7200.00	4632.94	0.28	28.21
gya1	18	3513204.32	3522650.84	0.27	7200.00	3518302.79	0.15	53.45
gya1	19	3501266.20	3509894.00	0.25	7200.00	3502926.68	0.05	115.24
gya1	20	3498654.87	3500110.57	0.04	7200.00	3499826.97	0.03	131.71
gya1	21	3500566.31	3500566.31	0.00	4194.36	3500566.31	0.00	95.69
gya1	22	3510439.96	3510439.96	0.00	4944.07	3510439.96	0.00	67.28
gya1	23	3532049.19	3532374.32	0.01	7200.00	3532288.28	0.01	82.74
gya1	24	3554955.43	3554955.43	0.00	1935.15	3554955.43	0.00	64.27
gya1	25	3585194.49	3585194.49	0.00	803.02	3585251.36	0.00	55.72
gya1	26	3622654.82	3622654.82	0.00	1113.47	3622654.82	0.00	54.34
gya1	27	3663910.96	3663910.96	0.00	343.29	3663910.96	0.00	54.92
gya1	28	3716663.42	3716663.42	0.00	591.89	3716714.23	0.00	43.74
gya1	29	3773370.06	3773370.06	0.00	405.58	3773371.19	0.00	50.81
gya1	30	3831365.77	3831365.77	0.00	120.78	3831393.55	0.00	45.43

3.4 比较分析

比较 SSCFLP 和 CFLSAP 计算结果，具有以下特征：(1)SSCFLP 增加设施服务区空间连续约束，目标值略有增加，ZY 案例增加 0.09%~1.00%，平均 0.39%；GY 案例增加 0.01%~ 0.36%，平均 0.13%。(2)SSCFLP 增加空间连续约束后，CPLEX 求解时间显著增加，求解质量也有下降，目标值与下界差异 MIPGap 从 0.00%增加到 0.20%。(3) SSCFLP 增加空间连续约束，对数学启发算法计算时间无明显影响。(4) SSCFLP 增加空间连续约束对设施区位和服务区有显著影响。本文测试案例中，部分案例无论是否考虑服务区空间连续，选择设施区位完全相同，但另一部分案例，SSCFLP 和 CFLSAP 区位选择有显著差异。例如，案例 zya4，图 2 为 SSCFLP 和 CFLSAP 计算结果示意图。两者目标值差异 0.39%，设施数量均为 19，但 3 个设施区位有差异；前者 4 个设施的服务区不连续，后者保证所有服务区空间连续。

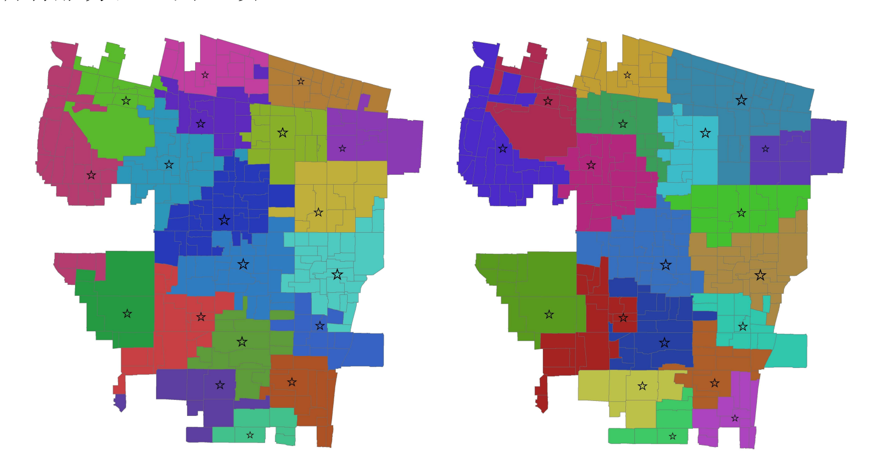
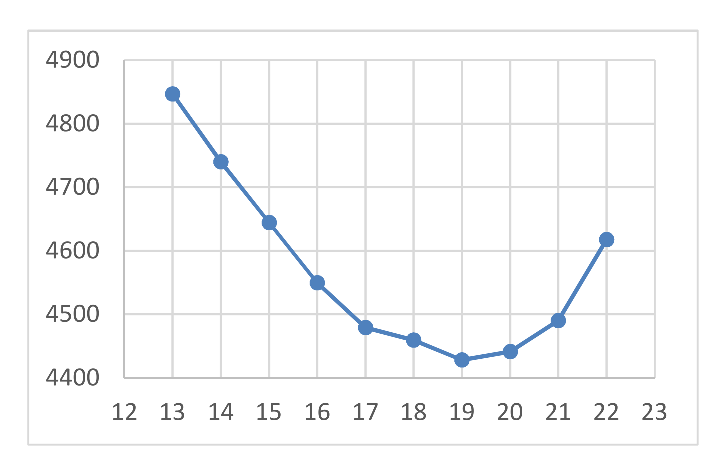


图 2 案例 zya4 计算结果(左为 SSCFLP，右为 CFLSAP)

Figure 2 SSCFLP (left) and CFLSAP (right) solutions on the instance zya4

设施数量约束对于案例目标值影响显著。首先，无论 SSCFLP 或 CFLSAP 问题，增加设施数量约束后，精确算法的求解时间显著下降。可能的原因是：增加设施数量约束后，可行的候选设施组合数量大幅降低，大幅缩小了可行解空间。CPLEX 计算结果表明，增加设施数量约束后，SSCKFLP 比 SSCFLP 更容易获得最优解，CKFLSAP 比

CFLSAP 容易获得最优解。其次， SSCFLP 或 CFLSAP 存在最优设施数量，增加设施数量约束，设施数量偏离原问题最优设施数量越多，目标增加值越多。图 3 为 SSCKFLP 目标值随设施数量变化曲线，上图为案例 zya1，下图为案例 gya1，两个案例 SSCFLP 最好解中，设施数量分别是 19 和 20。



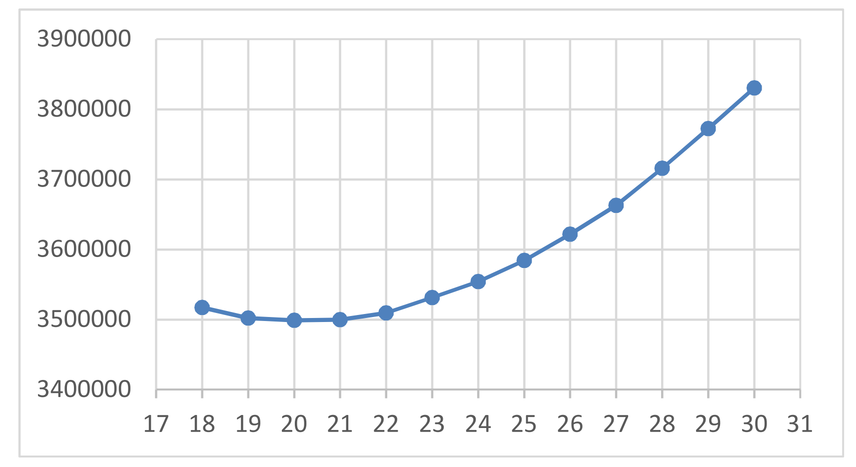


图 3 SSCKFLP 目标值(纵轴)随设施数量(横轴)变化曲线

Figure 3 SSCKFLP objectives on instances zya1 (upper) and gya1 (lower) versus the number of facilities

在需求分布确定、候选设施位置确定的情况下，设施容量和固定成本对区位选择有一定的影响。本文案例中，按同一比例调整设施容量，设施区位选择通常会发生变化。而同一比例调整设施成本，部分案例的区位选择仍保持不变。比较发现，以下各组案例的设施区位选择完全相同：zya1、zya2 和 zya3，zyb1、zyb2、zyb 3、zyb 4 和 zyb 5，zyc1、zyc2、zyc3、zyc4 和 zyc5，gya2 和 gya3, gya4 和 gya5，gyb2 和 gyb3，gyc1、gyc2 和 gyc3。

4 结论

本文对经典的 SSCFLP 进行了扩充，即增加设施服务区空间连续约束和设施数量约束，衍生出 SSCKFLP、CFLSAP 和 CKFLSAP 这 3 个变种问题。利用设施区位，借鉴网络流模型(Duque et al. 2011)，作者构建了顾及空间连续的 CFLSAP 和 CKFLSAP 模型。

使用流模型构建 p-regions 区划问题模型，模型精确求解仅适用于规模很小的案例；而本文 CFLSAP 和 CKFLSAP 模型相对容易求解，原因在于 SSCFLP 选择的设施区位可以直接用于建立流模型。实验表明：CPLEX 优化器能够求解 1276 个单元和 33 个设施的案例，获得最优解或高质量解(MIPGap< 0.44%)。SSCFLP 应用实践中，可能要求设施服务区连续，也可能需要指定设施数量，本文 SSCFLP 扩展模型符合这些应用需求。

本文设计了数学启发算法求解 SSCFLP、SSCKFLP、CFLSAP 和 CKFLSAP。数学启发算法本质上是超大规模邻域搜索，即从初始解作开始，使用数学模型迭代地改进当前解中的超大国模邻域，直到当前解难以更新。案例测试表明，数学启发算法求解 SSCFLP、SSCKFLP、CFLSAP 和 CKFLSAP，目标值与其下界的差异分别为 0.01%、0.00%、0.22%和 0.08%，求解质量与 CPLEX 计算结果无明显差异，但计算效率大幅提升。案例测试验证了本文数学启发算法的有效性。

本文案例研究也发现：SSCFLP 或 SSCKFLP 增加设施服务区空间连续约束后，目标值增加幅度不大，但最优设施区位可能发生变化。SSCFLP 或 CFLSAP 增加设施数量约束对于目标值影响显著，设施数量偏离原问题最优设施数量越多，目标增加值越多。在需求分布确定、候选设施位置确定的情况下，设施容量和固定成本对区位选择有一定的影响。

参考文献

[1] Aardal K, van den Berg P L, Gijswijt D, et al. Approximation algorithms for hard capacitated k-facility location problems. European Journal of Operational Research, 2015, 242(2):358-368.

[2] Agar M, Salhi S. Lagrangean heuristics applied to a variety of large capacitated plant location problems. Journal of the Operational Research Society, 1998, 49:1072–1084.

[3] Ahuja R K, Orlin J B, Sharma D. Multi-exchange neighborhood search algorithms for the capacitated minimum spanning tree problem. Mathematical Programming, 2001, 91:71–97.

[4] Avella P, Boccia M. A cutting plane algorithm for the capacitated facility location problem. Computational Optimization and Applications, 2009, 43 (1): 39-65.

[5] Barceló J, Casanovas J. A heuristic Lagrangean algorithm for the capacitated plant location problem. European Journal of Operational Research, 1984, 15(2):212–226

[6] Barceló J, Fernández E, Jörnsten K. Computational results from a new Lagrangean relaxation algorithm for the capacitated plant location problem. European Journal of Operational Research, 1991, 53(1):38–45.

[7] Basu S, Sharma M & Ghosh P S. Metaheuristic applications on discrete facility location problems: a survey. OPSEARCH, 2015, 52, 530–561.

[8] Beasley J E. Lagrangian heuristics for location problems. European Journal of Operational Research, 1993, 65:383–399.

[9] Beasley J E. OR-Library: Distributing test problems by electronic mail. Journal of the Operational Research Society, 1990. 41:1069–1072.

[10] Boccia M, Sforza A, Sterle C, Vasilyev I. A cut and branch approach for the capacitated p-median problem based on Fenchel cutting planes. Journal of Mathematical Modelling and Algorithms, 2008, 7(1):43–58.

[11] Caserta M & Voß S. A general corridor method-based approach for capacitated facility location. International Journal of Production Research, 2020, 58:13: 3855-3880.

[12] Chen C.H., C.J. Ting. Combining lagrangian heuristic and ant colony system to solve the single source capacitated facility location problem. Transportation Research Part E: Logistics and Transportation Review, 2008, 44 (6): 1099-1122.

[13] Contreras I A, Díaz J A. Scatter search for the single source capacitated facility location problem. Annals of Operations Research, 2008, 157 (1):73-89.

[14] Cornuejols G, Sridharan R, Thizy J. A comparison of heuristics and relaxations for the capacitated plant location problem. European Journal of Operational Research, 1991, 50(3):280–297.

[15] Cortinhal M J, Captivo M E. Upper and lower bounds for the single source capacitated location problem. European Journal of Operational Research, 2003, 151(2): 333-351.

[16] Duque J C, Church R L, Middleton R S, et al. The p-Regions problem. Geographical Analysis, 2011, 43(1): 104-126.

[17] Delmaire H, Díaz J A, Fernández E, Ortega M. Reactive GRASP and tabu search based heuristics for the single source capacitated plant location problem. Infor-Information Systems and Operational Research, 1999, 37(3): 194-225.

[18] Díaz J, Fernández E. A branch-and-price algorithm for the single source capacitated plant location problem. Journal of the Operational Research Society, 2002, 53:728–740.

[19] Filho V J M F, Galvao R D. A tabu search heuristic for the concentrator location problem. Location Science, 1998, 6: 189–209.

[20] Gadegaard S L, Klose A, Nielsen L R. An improved cut-and-solve algorithm for the single-source capacitated facility location problem. EURO Journal on Computational Optimization, 2018, 6: 1–27.

[21] Guastaroba G, Speranza M G. A heuristic for BILP problems: The Single Source Capacitated Facility Location Problem, European Journal of Operational Research, 2014, 238 (2): 438-450.

[22] Hindi H, Pieńkosz K. Efficient solution of large scale, single-source, capacitated plant location problems. Journal of the Operational Research Society, 1999, 50:268–274

[23] Holmberg K, Rönnqvist M, Yuan D. An exact algorithm for the capacitated facility location problems with single sourcing. European Journal of Operational Research, 1999, 113 (3): 544-559.

[24] Holt J, Ronnqvist M, Tragantalerngsak S. A repeated matching heuristic for the single-source capacitated facility location problem. European Journal of Operational Research, 1999, 116:51–68.

[25] Klincewicz J., Luss H. A Lagrangean relaxation heuristic for capacitated facility location with single-source constraints. Journal of the Operational Research Society, 1986, 37:495–500.

[26] Klose A, Görtz S. A branch-and-price algorithm for the capacitated facility location problem. European Journal of Operational Research, 2007, 179(3):1109–1125.

[27] Kong Y F, Zhu Y F, Wang Y J. A hybrid metaheuristic algorithm for the school districting problem. Acta Geographica Sinica, 2017, 72(02):256-268. [孔云峰, 朱艳芳, 王玉璟. 学校分区问题混合元启发算法研究[J]. 地理学报, 2017, 72(2): 256-268.]

[28] Neebe A W, Rao M R. An algorithm for the fixed-charge assigning users to sources problem The Journal of the Operational Research Society, 1983, 34:1107–1113.

[29] Oliveira Ó, Matos T, Gamboa D. A RAMP Algorithm for Large-Scale Single Source Capacitated Facility Location Problems. In: Matsatsinis N, Marinakis Y, Pardalos P. (eds) Learning and Intelligent Optimization. LION 2019. Lecture Notes in Computer Science, 2020, vol 11968, Springer, Cham.

[30] Rönnqvist M, Tragantalerngsak S, Holt J. A repeated matching heuristic for the single-source capacitated facility location problem. European Journal of Operational Research, 1999, 116(1):51–68.

[31] Shirabe T. Districting modeling with exact contiguity constraints. Environment and Planning B: Planning and Design, 2009, 36(6): 1053-1066.

[32] Sridharan R. A Lagrangian heuristic for the capacitated plant location problem with single source constraints. European Journal of Operational Research, 1993, 66(3):305–312.

[33] Tran T H, Scaparra M P, O'Hanley J R. A hypergraph multi-exchange heuristic for the single-source capacitated facility location problem, European Journal of Operational Research, 2017, 263(1): 173-187.

[34] Ulukan Z, Demircioğlu E. A Survey of Discrete Facility Location Problems. International Journal of Industrial and Manufacturing Engineering, 2015, 9(7): 2487-2492.

[35] Wang, Y., Xu, D., Du, D. et al. An approximation algorithm for k-facility location problem with linear penalties using local search scheme. J Comb Optim 36, 264–279 (2018). https://doi.org/10.1007/s10878-016-0080-2

[36] WANG W. A Local Search Algorithm for Hard Uniform Capacitated k-facility Location Problem. Nanjing Normal University, 2017.

[37] Yang Z, Chu F, Chen H. A cut-and-solve based algorithm for the single-source capacitated facility location problem. European Journal of Operational Research, 2012, 221 (3): 521-532.

A matheuristic algorithm for the single-source capacitated facility location problem and its variants

Abstract: This article proposes a matheuristic algorithm for the single-source facility location problem (SSCFLP) and its variants: SSCFLP with K facilities (SSCKFLP), SSCFLP with connective service areas (CFLSAP), and SSCFLP with K facilities and connective service areas (CKFLSAP). The algorithm starts from an initial solution, and then iteratively improves the solution by searching large-scale neighborhood of current solution. The neighborhood is defined by determining a subset of candidate facilities and a subset of customers: (1) randomly select a customer; (2) select Q nearest facilities and their customers from the current solution; (3) select nearest candidate facilities of all the customers; and (4) randomly drop some candidate facilities if too many facilities are selected in previous step. The size of neighborhood is critical to the performance of the algorithm: it is hard to solve an extra-large neighborhood and it is difficult to find a better solution in a small neighborhood. The value of Q is suggested according to the number of facilities in current solution. The current solution might be improved by finishing the following steps: (1) formulate a sub-problem model using the selecting facilities and customers; (2) solve the model and update the current solution using sub-problem solution; and (3) for CFLSAP or CKFLSAP, repair the non-connective service areas, and improve solution with local search operators. Two set of instances were generated to test the algorithm. Experimentation shows that the instances of SSCFLP and its variant problems can be solved by the proposed matheuristic algorithm effectively and efficiently. The solutions found by the proposed algorithm approximate optimal solutions or the lower bounds with average gaps of 0.01% for SSCFLP, 0.22% for CFLSAP, 0.00% for SSCKFLP, and 0.08% for CKFLSAP. Solution results show that the solution objective would be slightly increased by adding the contiguity constraints on SSCFLP or SSCKFLP. The optimal facility locations of SSCFLP/SSCKFLP might be different from those of CFLSAP/CKFLSAP.

Keywords: single-source capacitated facility location problem; connective service area; matheuristic algorithm; Large-scale neighborhood.

Funding: This work is supported by the National Natural Science Foundation of China (41871307).